Epistemic Landscapes Reloaded: An Examination of Agent-Based Models in Social Epistemology

Fernández Pinto, Manuela; Fernández Pinto, Daniel

Historical Social Research
Historische Sozialforschung

Manuela Fernández Pinto & Daniel Fernández Pinto:

Epistemic Landscapes Reloaded: An Examination of Agent-Based Models in Social Epistemology.

doi: 10.12759/hsr.43.2018.1.48-71

Published in:

Historical Social Research 43 (2018) 1

Cite as:

Fernández Pinto, Manuela, and Daniel Fernández Pinto. 2018. Epistemic Landscapes Reloaded: An Examination of Agent-Based Models in Social Epistemology. *Historical Social Research* 43 (1): 48-71. doi: 10.12759/hsr.43.2018.1.48-71.

Historical Social Research
Historische Sozialforschung

All articles published in HSR Special Issue 43 (2018) 1: Agent-Based Modeling in Social Science, History, and Philosophy.

Epistemic Landscapes Reloaded: An Examination of Agent-Based Models in Social Epistemology

Manuela Fernández Pinto & Daniel Fernández Pinto *

Abstract: *»Epistemische Landschaften reloaded. Eine Analyse agentenbasierter Modelle in der sozialen Erkenntnistheorie«.* Weisberg and Muldoon's epistemic landscape model (ELM) has been one of the most significant contributions to the use of agent-based models in philosophy. The model provides an innovative approach to establishing the optimal distribution of cognitive labor in scientific communities, using an epistemic landscape. In the paper, we provide a critical examination of ELM. First, we show that the computing mechanism for ELM is correct insofar as we are able to replicate the results using another programming language. Second, we show that small changes in the rules that determine the behavior of individual agents can lead to important changes in simulation results. Accordingly, we claim that ELM results are robust with respect to the computing mechanism, but not necessarily across parameter space. We conclude by reflecting on the possible lessons to be gained from ELM as a class of simulations or cluster of models.

Keywords: Social epistemology, epistemic landscape, agent-based models, division of cognitive labor.

1. Introduction[1]

In order to study the social organization of epistemic systems, particularly the problem of finding an optimal division of cognitive labor, social epistemologists have taken advantage of the conceptual and methodological tools offered by other scientific disciplines. Initial attempts to address the problem of the division of cognitive labor used rational choice theory and analytic models to

* Manuela Fernández Pinto, Universidad de los Andes, Departamento de Filosofía, Cra 1 No. 18A-10, Bogotá, Colombia; m.fernandezp@uniandes.edu.co.
Daniel Fernández Pinto, Calle 61 No 4-38, Bogotá, Colombia; dfernandezp@unal.edu.co.

1 The authors would like to thank Michael Weisberg and Ryan Muldoon for sharing their original code for ELM, and for answering our questions. Thanks also to the organizers and participants of the *I Congreso de Lógica y Epistemología Contemporánea* at the Universidad Nacional de Colombia, as well as the participants of the TINT Seminar at the University of Helsinki, for their valuable comments on previous drafts. Finally, special thanks to Jaakko Kuorikoski, Luis Mireles Flores, Samuli Pöyhönen, Petri Ylikoski, and two anonymous reviewers, for their thoughtful comments.

show that a scientist's non-epistemic interests, such as personal credit, can play a role in diversifying research efforts within the scientific community and thus can actually contribute to the success of science (Kitcher 1990; Goldman and Shaked 1991; Brock and Durlauf 1999; Strevens 2003). These rational choice models, however, have important limitations, especially when applied to questions of social epistemology and philosophy of science (Fernández Pinto 2016). In these models, scientists have perfect information about both the distribution of other scientists in different research projects and the probability of success of different research projects. If these conditions are relaxed, however, the models do not yield an optimal distribution of cognitive labor (Muldoon and Weisberg 2011). In addition, rational choice models of the division of cognitive labor work with homogenous individual agents, whose preferences are determined through an expected utility function. Results at the social level are thus the aggregate of individual preferences, and scientific success is at the end explained through an invisible hand mechanism: the optimal division of cognitive labor emerges unintentionally from the agents' individual preferences. Philosophers have questioned this particular use of invisible hand explanations, where the working of the invisible hand mechanism is taken for granted, as opposed to being empirically uncovered (Ylikoski 1995; Sent 1997).[2]

Given the limitations of rational choice models, social epistemologists have now opted for agent-based models and computer simulations to grant more complex dynamics of the division of cognitive labor (Hegselmann and Krause 2006; Weisberg and Muldoon 2009; De Langhe 2018 in this Special Issue). One advantage of agent-based models is the heterogeneity of agents, where every agent is represented explicitly in the model and where different types of agents are allowed. Instead of having perfect information and expected utility maximization, agent-based models are "bottom-up" in the sense that the behavior of the agents is generated by direct interaction in the model, instead of being imposed "top-down" by pre-established rules (Miller and Page 2007). In most cases, computer simulation models, such as agent-based models, are not analytical models in which equations are solved through mathematical manipulation with closed form solutions, but numerical models which are solved through an incremental time-stepping procedure iterated over time. Numerical models might not have closed form solutions, but numerical solutions or approximations, which go beyond human computing capability or their closed

[2] Note that this particular feature of early rational choice models in social epistemology might be overcome by the dynamics in social networks models (e.g., Zollman 2010), where the results at the social level are not just mere aggregations of individual preferences, but also depend on the network structure. In this sense, social network models might provide a better account of the underlying mechanism than early rational choice models, such as Kitcher's model.

form solution, might be just too difficult to find, so that we resort to computational aid.

One important contribution to addressing the problem of the division of cognitive labor using agent-based models is the epistemic landscape model (ELM) developed by Weisberg and Muldoon (2009). Mapping the microstructure of scientific research on an epistemic landscape in which the topology is given by the epistemic significance of research approaches, individual scientists explore the landscape according to differing interests and preferences. ELM then shows the efficacy with which different groups of scientists make epistemic progress and climb to peaks of epistemic significance. Looking at the progress of different agent distributions, ELM provides an innovative approach to establishing the optimal distribution of cognitive labor, one that accounts better for the heterogeneity of agents in real science than rational choice models, and one that allegedly avoids inadequate invisible hand explanations insofar as the simulations show the underlying mechanism at work.

The aim of this paper is twofold. First, we show that the computational mechanism of ELM is correct insofar as we are able to replicate the results using another programming language. Second, we show that small changes in the rules that determine the behavior of individual agents can lead to important changes in simulation results. Accordingly, we claim that ELM results are robust with respect to the computing mechanism, but not necessarily across parameter space. We conclude by reflecting on the possible lessons to be gained from ELM as a class of simulations, or cluster of models, and we suggest some possible directions to move forward.

The paper is divided in six sections. In the second section we present a brief description of the ELM model and the main results obtained by Weisberg and Muldoon. After reviewing Muldoon's (2007) argument for robust simulations, in the third section we explain our technical reasons for attempting a replication of ELM in FreePascal. The fourth section describes the results of our simulations (ELM-P). In the fifth section, we introduce two alternatives to Weisberg and Muldoon's Follow Rule, and we present the results obtained with these changes. Finally, in the last section we compare and discuss the results obtained using the original Follow Rule and the new alternatives, and we draw the conclusions of our analysis.

2. The Epistemic Landscape Model (ELM)

Weisberg and Muldoon (2009) introduced their ELM as an alternative approach to rational choice models for the study of the division of cognitive labor in social epistemology. In particular, they are not interested in modeling situations where scientists aim at a unique narrow goal using different approaches – i.e., the type of situation encapsulated in Streven's priority rule (2003) and frequently illustrated with the example of the race to find the structure of DNA – but instead, they aim at modeling the more common situation of scientists studying the same research topic using different approaches, without constraining themselves to a single research outcome, and where the achievement of epistemic success by some does not necessarily imply the failure of others.

In ELM then the division of cognitive labor is "represented as the distribution of agents throughout the landscape and scientific change as the exploration of the landscape" (2009, 228). An epistemic landscape or "grid" represents the particular research topic that interests a scientific community. A particular position in the landscape or "patch" represents the scientist's (or the research group's) particular "approach" to the study of the research topic in question. For Weisberg and Muldoon, an approach is composed of at least four aspects: (1) research questions, (2) instruments and techniques for data gathering, (3) methods for data analysis, and (4) background theories for data interpretation (2009, 228).[3] The combination of all the possible approaches across these four aspects makes then the epistemic landscape.[4]

Finally, the topography of the landscape is given by the epistemic significance of the approach. Following Kitcher's claim that science aims at *significant truths* (1993), the higher the significance of the results yielded by a particular approach, the higher the patch corresponding to such approach in the epistemic landscape.[5]

In addition, the authors introduce the notion of *epistemic progress* to acknowledge that the exploration of significant approaches that are not maximally significant also constitutes an important aspect of scientific research. Accordingly, epistemic progress is understood as "the percentage of patches with significance greater than zero that have been visited by the community of scientists" (2009, 237).

In an agent-based model such as ELM, individual scientists or research groups are explicitly represented as individual agents moving across the landscape. In contrast to rational choice models, ELM is able to model the behavior

---

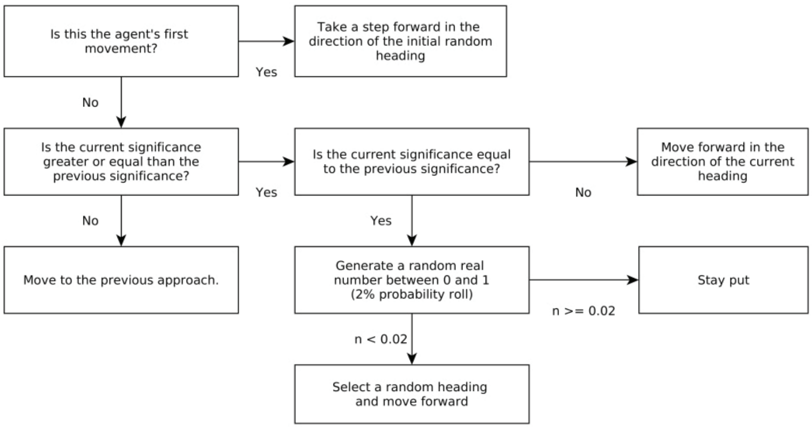
3 Weisberg and Muldoon recognize that approaches might have other non-epistemic components such as technological significance or monetary value (2009, 229, note 4). One can argue that even other epistemic components, such as experimental design, need to be added to the account as well.

4 As Weisberg and Muldoon duly note, if we were to take into account all the dimensions along which approaches may vary, the landscape would have high dimensionality. In favor of clarity and simplicity, ELM works with a three-dimensional landscape, where two dimensions correspond to aspects of the approach, such as the aspects 1-4 previously mentioned, and the third dimension corresponds to the epistemic significance of the results obtained following the approach.

5 Here the authors assume that a scientist or a research group that follow an approach that would in principle yield significant results would be in fact able to obtain such results. In this sense, the particular ability or talent of the scientist or the research group to develop the approach appropriately is not taken into account.

of different types of agents, who move in the landscape according to different rules. Weisberg and Muldoon (2009) specify the rules that determine the behavior of three different types of agents. The HE Rule fixes the behavior of *control agents*, who move across the landscape in the simplest way, starting with a population of agents randomly distributed in zero-significance zones. Figure 1 shows the flowchart according to which control agents decide how to move.

Figure 1: Flowchart Representing the Coding of the HE Rule in ELM



Controls only take into account the epistemic significance of the patches in their close neighborhood,[6] in order to decide whether to move or not and in which direction. But of course, individual scientist or research groups do not explore approaches to a research topic completely in isolation, as controls do. In contrast, they learn from what their fellow scientists have done and make decisions accordingly. Assuming that the agents will have information about the patches in their neighborhood that have been already explored, Weisberg and Muldoon introduce two additional types of agents. *Followers*, or agents that tend to move to previously explored patches, and *mavericks*, or agents who tend to look for new unexplored patches in the landscape.

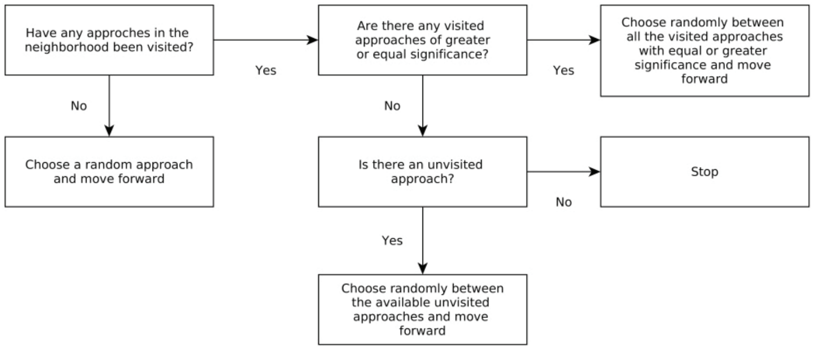
Followers simulate conservative scientists who are more comfortable guiding their research according to already explored approaches. According to the programming code, followers in ELM first evaluate whether any of the patches in their neighborhood has been explored.[7] If any has been, then they evaluate

[6] A neighborhood in ELM is equivalent to a Moore neighborhood or the eight patches surrounding the patch where the agent is located.

[7] We found a discrepancy between the followers rule that Weisberg and Muldoon present in the paper and the followers rule coded in their NetLogo model. While the paper version re-

whether the significance of the patch is greater or equal to the significance of their current patch, and move accordingly. Figure 2 presents the decision-making process of followers as coded in ELM.

Figure 2: Flowchart Representing the Coding of the Follow Rule in ELM

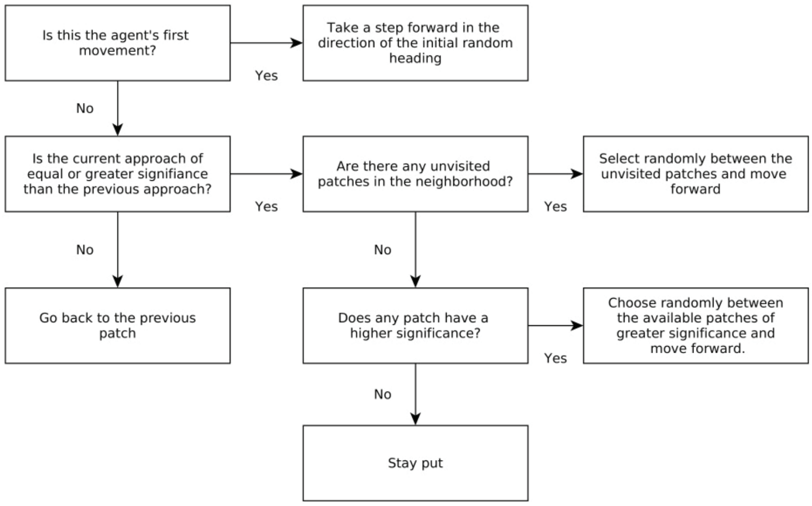


Mavericks also take into account what their fellow scientists have done, but unlike followers, they prefer to move to unexplored patches in the landscape. They simulate scientists who are risk-takers. Accordingly, they first evaluate whether the significance of the patches in their neighborhood is greater than the significance of their current patch and if so, then they evaluate whether it has been already explored, moving always to unexplored patches when available. Figure 3 presents the decision-making process of mavericks coded in ELM.

In their 2009 paper, Weisberg and Muldoon report having run simulations on controls for 50,000 cycles starting with 10 agents with an increment of 10 agents at a time up to a total of 400 agents. For each group of agents (10, 20, 30, etc.), they ran 100 simulations. They ran similar simulations with followers, but only through 1,000 cycles, and with mavericks, through 200, 500, and 2,000 cycles. Finally they also ran simulations of two types of mixed populations, one with 400 followers and only one maverick through 1,000 cycles, and a second one with an increasing population from 10 to 400 followers, where mavericks were introduced 10 at a time, up to 50, through 500 cycles.

stricts the movement of followers exclusively to patches of greater significance, the code allows followers to move when the available patches have greater or equal significance. Alexander et al. (2015) have also identified the same discrepancy. As it will become clear in the following sections, this small difference in the specification of the rule yields significantly different results.

Figure 3: Flowchart Representing the Coding of the Maverick Rule in ELM



The results of the simulations are given in terms of the epistemic progress of the agents. Given pure populations of controls, followers, and mavericks, Weisberg and Muldoon conclude that mavericks are most efficient at both finding peaks and making epistemic progress, whereas followers are the least efficient in both tasks. Controls do slightly better than followers both in finding peaks and in exploring significant zones (2009, 245, Figure 9). In the end, they conclude that mixed populations are ideal, where a small number of mavericks can find epistemic peaks quickly and significantly stimulate the epistemic progress of followers, while followers are good at exploring the breadth of the epistemic landscape (2009, 250).

3. Replication in a Different Programming Language

As previously noted, one of the main motivations for developing agent-based models of the division of cognitive labor, and ELM in particular, is trying to capture the inherent complexity of the social organization of scientific communities that analytic models fail to capture and for which computer simulations seem a promising tool. Analytic models, however, have been praised precisely for their simplicity and clarity, as well as for the modeler's control over the equations and mathematical calculations of the model. Agent-based models on the contrary often contain stochastic components that generate complex non-linear behavior, which are not trivial for human beings to solve analytically and thus require computational simulations. Given the modelers' inability to check the calculations, one could in principle object that there is no way to know whether the results are correct or whether there is an error in the simulation.[8]

Appealing to robustness analysis, Muldoon (2007) argues against this objection. He acknowledges that agent-based models are prone to computer failures that analytic models are not, and he identifies software errors – mistakes or unknown implementation details in the programming language or software library – as one of this type of failures (2007, 877). However, Muldoon argues that robustness analysis is an adequate way of overcoming these limitations. Given that there are a number of programming languages available, replicating simulation results in different programming languages would decrease the likelihood of programming errors, making the simulation results robust across different programing languages.

Weisberg and Muldoon (2009) initially programmed the ELM in NetLogo, which is a program intended especially for agent-based model simulations. NetLogo is considered an *interpreter* as it executes code from a scripting language designed specifically for the program. This allows for the easy coding of agent-based models without the need to code from a lower level of abstraction. Despite the many advantages of NetLogo, one might wonder, precisely because of the built-in functions of the program, whether the ELM is prone to the software errors that Muldoon (2007) identifies.

Following the argument for robustness analysis, we decided to program the ELM model in FreePascal in order to overcome possible software failures associated with the use of a higher abstraction language such as NetLogo. The FreePascal programming language and compiler – a more recent open source derivative of Apple's proprietary Pascal language and compiler implementations – does not rely on any interpreter and therefore all aspects of the code can be tightly controlled by the programmer. This allowed us to develop a modeling framework that was flexible, enabling a large array of potential modifications which would be difficult to implement in a more rigid environment such as NetLogo.

Using FreePascal has also other advantages. It is faster than languages such as NetLogo, while being also a cross-platform compatible language. In addition, thanks to the open source Lazarus IDE, rich user-interface applications are easy to develop in FreePascal. We realize that other programming languages (C, Python, Java) might also be fit for the coding and analysis of this model, and that compiler or even hardware related bugs might still be present, regardless of the level of abstraction of the programming language involved.

The ELM and ELM-P (ELM model coded in FreePascal) have some differences inherent to the way in which both implementations were coded. In NetLogo agents have a given heading and their movement is controlled by

[8] For further discussion about this phenomenon, also known as 'epistemic opacity,' see Humphreys (2004, 147-51).

giving stay-forward-backwards or change-heading instructions, while in our model agents do not have a predetermined heading but evaluate their neighborhood on each time step in order to make a decision to move to a new patch or stay on their current one. This creates a clear difference in coding structure which does not translate into relevant qualitative differences within our results. We made this coding decision as the addition of a heading variable was possible but not needed in FreePascal, while in NetLogo it's an inherent property of the agent being used and therefore can be used with no additional programming effort.

Another important difference between both implementations is the random number generator. Both FreePascal and Netlogo use the Mersenne twister algorithm for random number generation[9] but their specific implementations are different. Although both random number generators have a good enough quality and should mimic the output of a non-deterministic process, differences between both implementations could arise because of this factor.

4. Results in ELM-P

We have adapted the HE Rule in the ELM model to our programming structure, doing an evaluation of all neighborhood patches in analogy to the process used within the Follow and Maverick Rules. A flowchart describing our adaptation of the HE Rule is shown in Figure 4.

Despite the evident differences in programming structure when comparing Figure 4 with Figure 1, we can still observe qualitatively similar results to those obtained by Weisberg and Muldoon (2009) when evaluating epistemic progress as a function of control agent and cycle number (Figure 5).

[9] Netlogo random number generator: <https://ccl.northwestern.edu/netlogo/docs/programming.html>. FreePascal random number generator: <http://www.freepascal.org/docs-html/rtl/ system/random.html> (both accessed January 16, 2018).

Figure 4: Flowchart Describing the Implementation of the HE Rule in the ELM-P Model

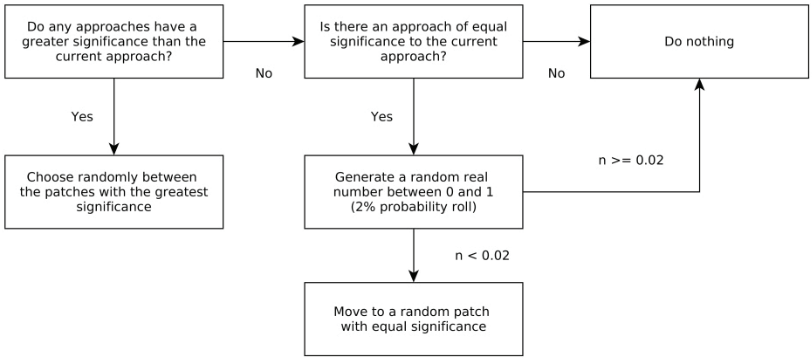
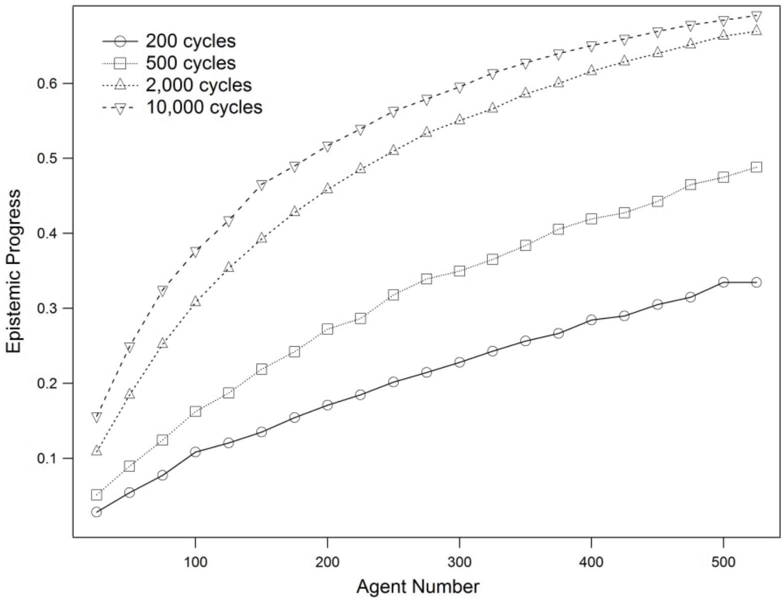
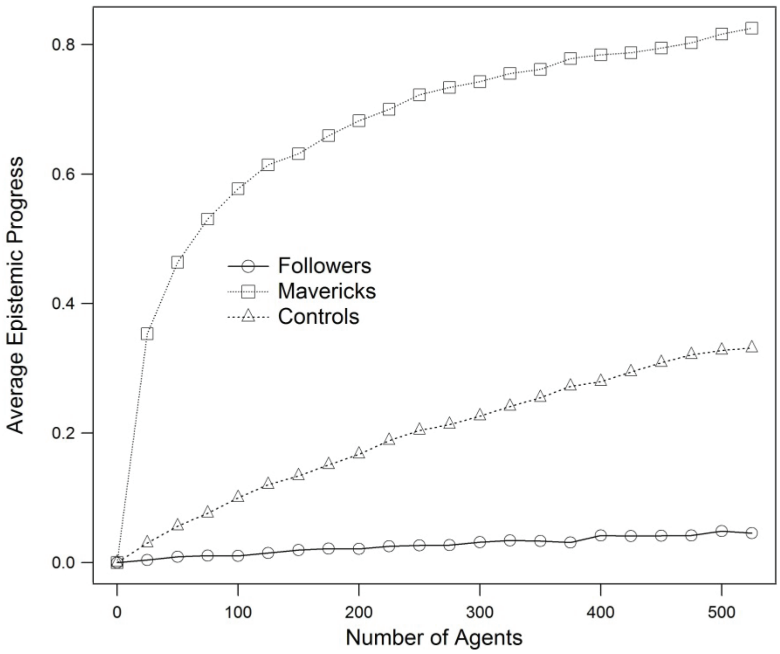


Figure 5: Epistemic Progress as a Function of Control Agent Number for Different Simulation Cycle Numbers



Epistemic progress increases as a function of agents and as a function of cycles, with the largest epistemic progress being achieved with the largest number of cycles and agents (550 agents, 10,000 cycles). Our final epistemic progress values at 400 agents are also quantitatively similar to those published within their paper for all cycle values. However it is worth noting that our curves at 2,000 and 10,000 cycles are notably more logarithmic in shape, pointing to a faster ability of our controls to reach the peaks within the landscape, leading to a lower final epistemic progress (as controls that reach the peaks remain static). The ability to act after evaluating their entire neighborhood makes the controls more efficient than those acting only on information obtained from their last movement. These differences are however small and do not constitute a dramatic difference with the controls in the original paper.

Figure 6: Comparison between Controls, Mavericks, and Followers for a 200 Period Cycle for Different Agent Numbers

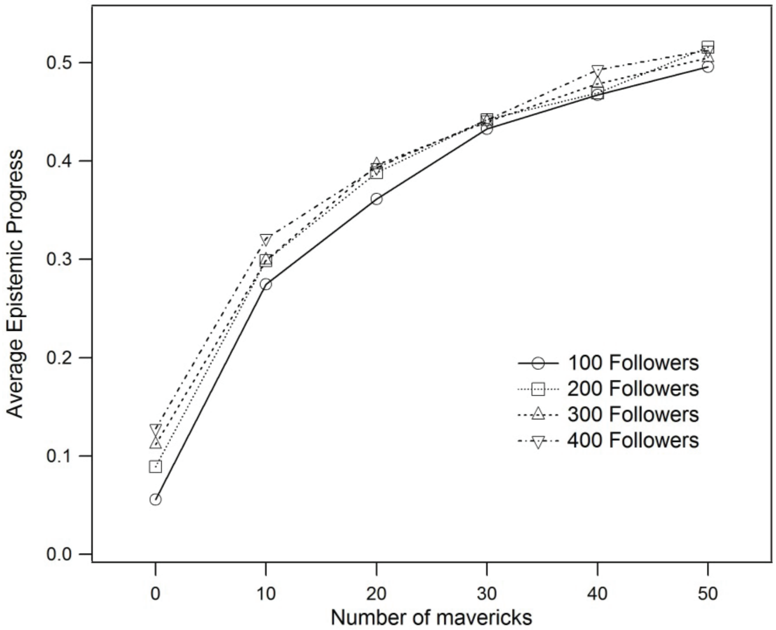


Our agents using the Follow Rule behave in a very similar manner to the original followers in ELM. Giving priority to spaces of equal or greater significance creates a problem: all followers that are not initially located next to a zone of non-zero epistemic significance become stuck in a loop between their initial and subsequent positions.[10] This is the main reason followers make very low epistemic progress, even with large number of agents. Figure 6 shows the comparison between followers, mavericks, and controls for a 200 period cycle length.

[10] This problem has also been noticed by Alexander et al. (2015).

In our results mavericks are also by far the most efficient agents at making epistemic progress. Figure 6 shows the same relationship obtained by Weisberg and Muldoon where epistemic progress is largest for mavericks followed by controls and finally followers. Our mavericks also reach similar epistemic progress values at 400 agents, with an average epistemic progress of almost 80% of the available non-zero epistemic landscape. Clearly our results are not equal to the results obtained by Weisberg and Muldoon as is expected from the programming implementation differences outlined in section 3. Nonetheless our results lead to the same conclusions and are quantitatively similar, as one can see when comparing their results (Weisberg and Muldoon 2009, 245, Figure 9) to ours (Figure 6).

Figure 7: Epistemic Progress of Mixed Communities after 500 Cycles



Finally, we also obtained similar results when dealing with mixed population of followers and mavericks. Weisberg and Muldoon ran simulations of 100-400 followers, adding populations of 10-50 mavericks at a time, for 500 cycles. As expected, populations with a higher number of mavericks attain epistemic progress faster, and the addition of even a small number of mavericks increases dramatically the epistemic progress of the community. Figure 7 shows the behavior of mixed populations in ELM-P, which is fairly similar to the results obtained in ELM (for comparison, see Weisberg and Muldoon 2009, 247, Figure 10).

Given that the implementation in FreePascal rendered similar results to the original implementation in NetLogo, we conclude that the results of ELM are robust across different programing languages, i.e., that the computing mechanism is correct and results are not due to calculation errors.

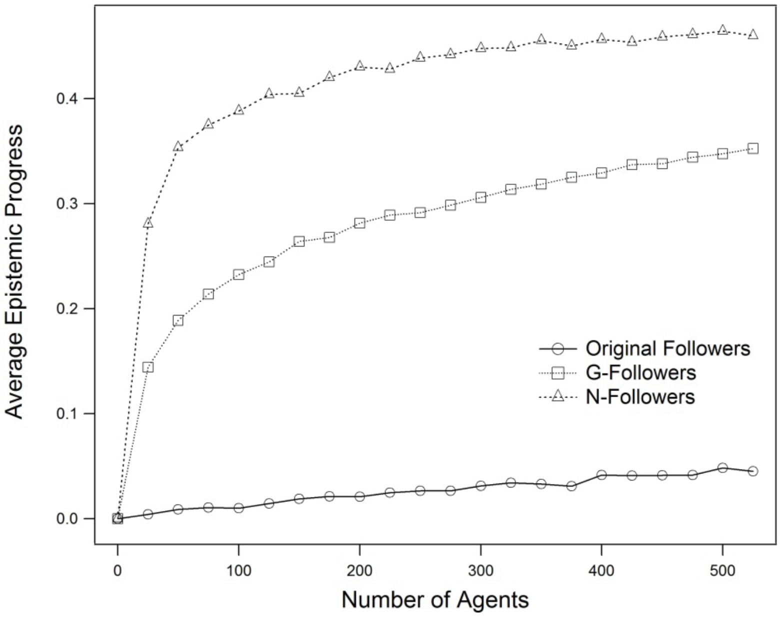
5. Alternative Follow Rules

So far, we have performed the robustness checks across programing languages to avoid computing problems in ELM. However, as Muldoon (2006, 880) recognizes, checking for robustness across programming languages is just the first step towards model validation in computer simulations: "One may be able to program a simulation in as many languages as one might please, but that does not help to ground the simulation, or its underlying model, to the real world in a meaningful way." Thus, in addition, we should run a stronger robustness analysis, i.e., "robustness across modalities and parameter space" (ibid.). Let us perform a couple of changes on the parameter space, accordingly, and observe the simulation results.

As we explained in Section 4 one of the main problems with the original Follow Rule is the use of an equal or greater comparison when evaluating where to move next among visited patches in the Moore neighborhood. If we remove this restriction – by moving only to visited spaces of *greater* significance – our followers become dramatically more efficient at exploring the epistemic landscape. We call this new implementation of the rule guiding the behavior of the followers the G-Follow Rule, given that followers are modified to use a *greater than* instead of an *equal or greater* comparison when selecting where to move, (in this way avoiding their return to the last explored patch). Although our followers are still significantly less efficient than mavericks they do become more efficient than the original followers and even the original control group.

Another alternative implementation of the rule guiding the behavior of followers is the N-Follow Rule, according to which followers move to explored spaces with higher or equal significance as in the original Follow Rule, but NEVER to their own previously explored spaces. The N-Follow Rule makes the agents even more efficient at exploring the landscape than the original followers, the g-followers, and the controls, as can be seen in Figure 8.

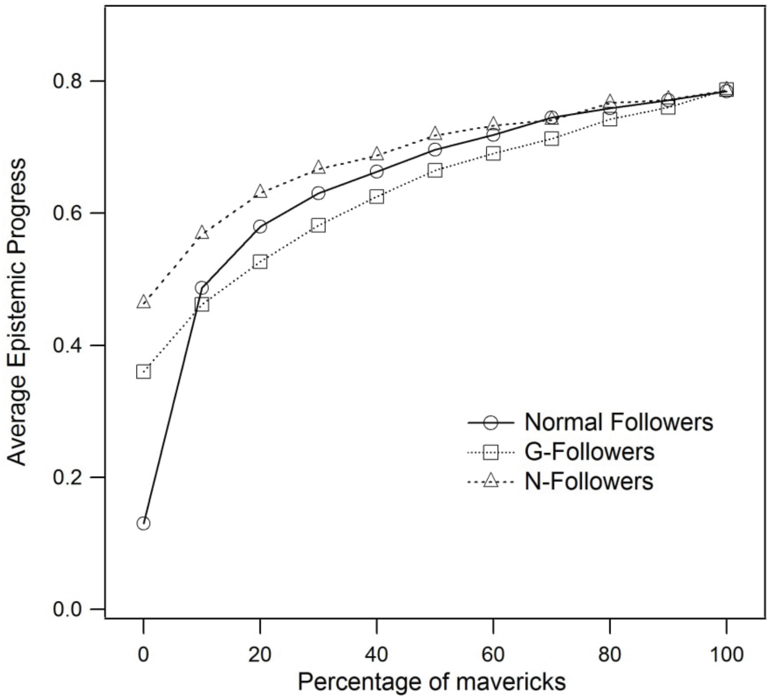
Figure 8: Average Epistemic Progress of the Original Follower, G-Followers, and N-Followers



The implementation of different versions of the Follow Rule also leads to different results in mixed populations. G-followers and n-followers make significantly more epistemic progress than the original followers when no mavericks or a small number of mavericks are present. Figure 9 illustrates such differences.

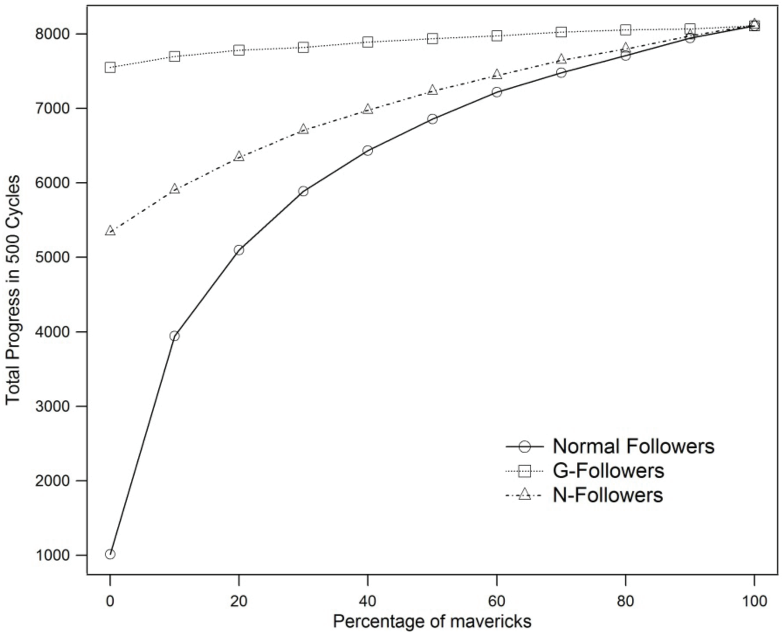
As the number of mavericks increases, the type of follower used in the simulation matters less and less, given that the epistemic progress is increasingly achieved by the mavericks. In other words, when the maverick population is small, 15% or less, the type of followers that we find in the epistemic community makes a difference in the community's epistemic achievements. This difference is important because mavericks are presumably more costly than followers (Weisberg and Muldoon 2009, 250), which encourages scientific communities to maintain the number of mavericks as low as possible. In this sense, the more epistemic progress achieved by a community with a low percentage of mavericks, the better.

Figure 9: Average Epistemic Progress of Mixed Populations of 400 Agents in 500 Cycles with Original Followers, G-Followers, and N-Followers



Another way of measuring the progress of the scientific community on ELM is to look at the number of approaches or patches explored in the landscape regardless of their significance. Weisberg and Muldoon call this a measure of *total progress*, since it gives us information about the overall activity of the scientific community, and not only of its success (2009, 248). In principle, followers make important contributions to the total progress of a community, given that they are much better at exploring Moore neighborhoods than mavericks, thus covering larger parts of the landscape faster. But here as well the type of follower that one is implementing in the model matters greatly (see Figure 10). On a landscape of 10,201 total approaches, as in ELM, the original followers contribute much less at exploring the landscape than n-followers or g-followers. G-followers in fact seem to contribute significantly to total progress, regardless of the number of mavericks in the community.

Figure 10: Total Progress of Mixed Populations of 400 Agents in 500 Cycles with Original Followers, G-Followers, and N-Followers

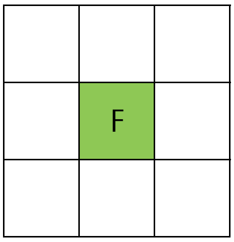


Such differences might have important consequences for the lessons drawn from the simulation. For instance, if one is implementing the original Follow Rule, one could say that the best way to obtain total progress in an area of scientific research is to have a mixed population of followers and mavericks with a higher percentage of mavericks. On the other hand, if one is implementing the G-Rule, one could claim that the number of mavericks only matters for epistemic progress, but not for total progress. Similarly, if one is implementing the original Follow Rule, one might claim that individual learning, i.e., the behavior exhibited by controls, is more efficient as a learning strategy than social learning, i.e., the strategy exhibited by followers, insofar as controls perform better than original followers (Pöyhönen 2016). However, if one implements g-followers or n-followers, then one must conclude just the opposite. Thus, differences in the Follow Rule implemented in the model can lead to different results, and radically different recommendations.

6. Discussion

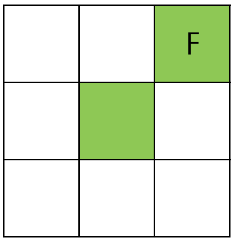
As we already mentioned, followers in ELM are supposed to learn from the achievements of other agents and use this information to find patches of greater significance. As Weisberg and Muldoon acknowledge, one of the limitations of the Follow Rule as implemented in ELM is that followers get stuck pretty easily. First, "if followers bump into each other on the way up, they can get stuck following each other around to a suboptimal region of the hill" (Weisberg and Muldoon 2009, 241-2). Second, if followers start far away from the hills, "they end up following their own trail. Or if around others, they end up circling around the trails each other make" (ibid., 242). We trace this particular limitation of the Follow Rule to the way it was originally implemented in ELM, where agents first evaluate whether any of the patches in their neighborhood have been explored, and if they have been, then they evaluate whether the significance of these patches is greater or equal to the significance of their current patch, and move accordingly. The problem with this implementation of the rule, as Alexander et al. (2015) have already noted, is that followers move to patches of great or *equal* significance, without distinguishing whether the previously explored patch has been explored by themselves or by other agents. Thus, the original Follow Rule allows the following move. Suppose there is a follower at a zero-significance zone surrounded by zero-significance patches as Figure 11a illustrates:

Figure 11a: Follower in Zero Significance Moore Neighborhood



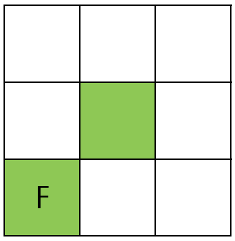
Since there aren't any significance patches in her Moore neighborhood, she randomly chooses to move, as follows:

Figure 11b: Follower Moving Randomly in a Zero-Significance Zone



Let’s assume that she is not close to a hill and, after the move, her new Moore neighborhood looks like this:

Figure 11c: Follower's New Moore Neighborhood



For the next move, the follower would identify the explored patch in her neighborhood and evaluate its significance. Since the significance is *equal* to the patch she is currently in (i.e., zero), then she would move back to this patch (see Figure 11c). In other words, the follower in this case would not be able to take into account that the explored patch in her Moore neighborhood has been explored *by her*, and she would easily get stuck back and forth between patches that only she has explored.

At first sight this does not look like a natural implementation of the Follow Rule. If the rule aims to model the behavior of scientists who learn from the successful work that other scientists have done, it would not make sense for followers to move back to their own unsuccessful approaches for solving a scientific problem.

One way to avoid the previous situation is to implement the rule allowing followers to move to previously explored patches only if the significance is greater than the significance of the patch where they currently are. This is the way we implemented the G-Follow Rule (see Section 5). The rationale behind the G-Follow Rule is that follower scientists would change their approach to research if they have access to other explored approaches that have already proved to be more successful. If there are other explored approaches that have not proved to be more successful, scientists would not have particular motivations to pick that approach among other available approaches. As we already mentioned, the implementation of the G-Follow Rule makes followers much more efficient not only than original followers, but also more efficient than control agents (see Figure 8, and compare with controls in Figure 6).

One could argue that the G-Follow Rule is also unnecessarily limited insofar as followers might have good reasons to move to other previously explored approaches even if they haven’t been more epistemically significant than their current approach. For example, scientists might argue that joining forces in exploring an approach might maximize the possibility of obtaining good results or they might find it convenient to adopt an already explored approach that is more suitable for their labs and expertise than the one they are currently developing. Thus, in principle there can be reasons for followers to move to explored patches even if they have equal significance to their current patch. In this sense, to assume that followers do not coordinate their movements with other agents and that their movements do not have any adaptation costs are substantial idealizations of the model that should be pointed out.

In order to avoid both problems, i.e., the problem of followers going back to patches that they have already explored and the problem of not allowing followers to move to already explored patches that are of equal significance to their current patch, we implemented the N-Follow Rule, where followers move to patches of equal or greater significance, but excluding their own previously explored patches (see Figure 8). As noted earlier, the N-Follow Rule makes followers much more efficient at exploring the landscape and making epistemic progress.

We think our results are coherent with the rationale that scientists building on research approaches that have already proved successful should make more epistemic progress than controls who explore the landscape randomly (cf., Pöyhönen 2016). It makes sense then for our g-followers to be more efficient than controls. Moreover, scientists who not only take advantage of what other scientists have achieved, but also avoid going back to research approaches that they already know to be unsuccessful should make more epistemic progress. It makes sense then for our n-followers to be more efficient than our g-followers. Accordingly, g-followers seem to capture better the behavior expected from scientists exploring a research topic than do the original followers, and n-followers do it even better than g-followers.

One point that becomes clear with the implementation of alternative rules for the behavior of followers is that small changes in the coding of the rules can result in major changes in the behavior of the agents. Notice, for instance, that the only change in the code that we made from the original Follow Rule to the alternative G-Rule was the change from accepting the move to patches of equal or greater significance (≥) to only accepting the move to patches of greater significance (>). This minor change in the implementation of the rule resulted in making g-followers around 10 times more efficient than their original counterparts. Similarly, by restraining followers from going back to their previously explored patches, n-followers became around 15 times more efficient than their original counterparts. Different types of followers also lead to different results when interacting with mavericks in mixed populations. Especially in mixed populations with a small percentage of mavericks, which type of Follow Rule is used seems to matter when accounting for the epistemic progress of the community. Moreover, the type of Follow Rule used also matters when accounting for the total progress of the community, regardless of the percentage of mavericks.

So what are the lessons to be learned from comparing the original results in ELM to the results obtained with changes to the Follow Rule in ELM-P? We could say that results regarding the performance of followers when compared to mavericks and controls in the original ELM are not robust, given that they do not hold across parameter changes.[11] A similar conclusion can be made regarding the contribution of followers to total progress, given that alternative followers significantly contribute to total progress even without mavericks, while this was not the case with original followers.

One result that seems to hold across parameter changes here is that mixed populations increase their efficiency with a higher percentage of mavericks, and mavericks do seem to provide pathways for followers to find zones of greater significance and thus boost followers' contribution to epistemic progress (Weisberg and Muldoon 2009, 250). However, in mixed populations with a small percentage of mavericks, our alternative followers contribute much more to epistemic progress than suggested by the original results, probably leading to different optimal balances between followers and mavericks within a scientific community.

Even though we only presented a couple of alternatives to the Follow Rule, the G-Rule, and the N-Rule, there are numerous ways to implement such a rule – other possible alternatives would include making followers wait in cycles in which they do not find explored patches around them, or letting them see patches beyond their Moore neighborhood (Weisberg 2013; Thoma 2015). Parameter changes with respect to the behavior of controls and mavericks should also be explored to better understand the mechanisms guiding the behavior of each type of agent, as well as the relation between different types of agents in ELM. Moreover, altering the topology of the landscape (Pöyhönen 2016), considering the capacity of the agents to explore each patch, or taking into account the risk of certain strategies, would all help exploring the parameter space of ELM and contribute to further understanding of the scope and fruitfulness of the model.

In this sense, even if some results are not robust across the parameter space in ELM, further exploration of the model, i.e., further alternation of parameters, could in principle lead us to uncover core mechanisms to understand the division of cognitive labor in scientific communities. If we understand ELM not as an individual model, but as a class of simulations (Muldoon 2006) or cluster of models (Ylikoski and Aydinonat 2014), which implement the same basic dy-

[11] We use the concept of robustness analysis instead of the more precise term *derivational robustness analysis* (Woodward 2006) to refer to the testing of the model across different parameter values. If the same conclusions are derived from such testing, then the results are considered to be *derivationally robust* (Kuorikoski and Lehtinen 2009; Ylikoski and Aydinonat 2010).

namic while altering different parameters at a time, then we could in principle obtain epistemically significant conclusions. In Muldoon's words:

> Our proper object of analysis is this class of simulations. To generate this class, we create multiple models that implement the same basic dynamic by different modalities and across parameter space. In doing so, we create independent pathways of investigation, which allows us to isolate any particular implementation detail by examining its effects on the simulation. (Muldoon 2006, 880)

Accordingly, one should be cautious about drawing any general lessons from an individual implementation of ELM. Agent-based models, such as ELM, are incredibly powerful and flexible tools for model building, but their advantages come at a cost: the models are not analytically tractable and they have a large parameter space (de Marchi and Page 2014). This means, among other things, that ABMs are highly manipulable, and thus it comes down to the modeler to justify her parameter choices and further explore the parameter space.

A careful exploration of the cluster of models could potentially lead to uncovering epistemically interesting results, for instance well-justified lessons regarding the mechanisms guiding the distribution of cognitive labor in scientific communities and possible ways to optimize it.[12] Such an understanding requires a systematic analysis of ELM:

> An ability to make reliable inferences about real-world systems presupposes a systematic understanding of the ways in which the changes in the assumptions of the model changes its results. (Ylikoski and Aydinonat 2014, 30)

In this sense, results obtained through implementing g-followers and n-followers in this paper, as well as other contributions to the exploration of the parameter space of ELM (Weisberg 2013; Alexander et al. 2015; Thoma 2015; Pöyhönen 2016), can be understood as contributing to this larger endeavor.

Robustness analysis however won't be enough. If ELM aims at capturing in any significant way the mechanisms that guide the division of cognitive labor in scientific communities in the real world, then even for a theoretical model such as ELM the parameters explored should be realistic or credible. Following Kuorikoski and Lehtinen,

> [i]f the substantial assumptions are not realistic, no amount of robustness analysis suffices to change our views about which results of the model could also be taken to hold in the real world. Robustness analysis is thus useless if all assumptions are unrealistic, and its epistemic relevance rides on there being at least some realistic assumptions. (2009, 127)

---

[12] Pöyhönen makes a similar point: "By tracing differences in outcomes to differences in modeling assumptions, the different models can together be seen to lead to a clearer picture of the potential and correct interpretation of epistemic landscape modelling – and more generally, to a better understanding of the possible mechanisms through which cognitive diversity influences the conduct of scientific research" (2016, 20).

Hence, in addition to the potential robust results that the exploration of the cluster of agent-based models may yield, we need a better way for deciding among the infinite possibilities of parameter space exploration, and in particular for deciding among possible rules of agent behavior. In this sense, rules that are more natural, more realistic, more credible, or portray a higher degree of empirical adequacy[13] seem more promising for the exploration of the parameter space than rules which seem utterly unrealistic, incredible, or just empirically inadequate. For instance, we consider that it is more natural to implement followers in ELM according to the G-Follow Rule than according to the original Follow Rule because we do not expect scientists to go back to research approaches that they have already tried unsuccessfully.

Of course, whether ELM is a fruitful model for capturing the mechanisms guiding the division of cognitive labor in science or not is an open empirical question. If it is, then further exploration of the cluster of models should generate some robust results, while the set of parameters to be explored remains credible.

7. Conclusion

Weisberg and Muldoon have made an important contribution to the study of the division of cognitive labor in social epistemology and philosophy of science, encouraging philosophers to explore epistemic problems through agent-based models. Our contribution to this effort in the paper is twofold. First, we show that the computational mechanism of ELM is correct insofar as we are able to replicate the results in FreePascal. Second, we show that small changes in the rules that determine the behavior of individual agents can lead to important changes in simulation results. Accordingly, we claim that ELM results are in general robust with respect to the computing mechanism, but not necessarily across parameter space. In other words, we find that with some small changes in the parameters, some of the results in ELM do not hold anymore.

A fruitful way to move forward is to understand ELM, and similar ABMs, as a class of simulations or cluster of models, requiring further exploration of the parameter space, which could lead potentially to finding robust results, and thus contribute to understanding the underlying mechanisms guiding the distri-

[13] We follow Longino (1995) in our preference of the term 'adequacy' over the more commonly used term 'accuracy' to describe the relation of agreement between the model and its target system, in this case, the scientific community. An ABM would have a high degree of empirical adequacy if, for example, it is properly parametrized and empirically validated. This is an important challenge for computer simulation models in social epistemology (Martini and Fernández Pinto 2017), given the empirical data available for such purpose. However, even if the models are theoretical and highly idealized, we still consider that the relevant parameters should be realistic if the models aim at capturing core mechanisms of the target system.

bution of cognitive labor in science. In addition to obtaining robust results, the exploration of the parameter space in ELM also requires the search for more credible or realistic assumptions, so that the model, even if highly idealized, is appropriately set to capture the mechanism guiding the distribution of cognitive labor in real-world science.

References

Alexander, Jason McKenzie, Johannes Himmelreich, and Christopher Thompson. 2015. Epistemic Landscapes, Optimal Search, and the Division of Cognitive Labor. *Philosophy of Science* 82 (3): 424-53.

Brock, William A., and Steven N. Durlauf. 1999. A Formal Model of Theory Choice in Science. *Economic Theory* 14: 113-30.

De Langhe, Rogier. 2018. An Agent-Based Model of Thomas Kuhn's Structure of Scientific Revolutions. *Historical Social Research* 43 (1): 28-47. doi: 10.12759/hsr.43.2018.1.28-47.

De Marchi, Scott, and Scott Page. 2014. Agent-Based Models. *Annual Review of Political Science* 17: 1-20.

Fernández Pinto, Manuela. 2016. Economics Imperialism in Social Epistemology: A Critical Assessment. *Philosophy of the Social Sciences* 46 (5): 443-72.

Goldman, Alvin, and Moshe Shaked. 1991. An Economic Model of Scientific Activity and Truth Acquisition. *Philosophical Studies* 63: 31-55.

Hartmann, Stephan. 1996. The World as a Process: Simulations in the Natural and Social Sciences. In *Modelling and Simulation in the Social Sciences from the Philosophy of Science Point of View, Theory and Decision Library,* ed. Rainer Hegselmann, Ulrich Mueller and Klaus G. Troitzsch, 77-100. Dordrecht: Kluwer.

Hegselmann, Rainer, and Ulrich Krause. 2006. Truth and Cognitive Division of Labor – First Steps towards a Computer Aided Social Epistemology. *Journal of Artificial Societies and Social Simulation (JASSS)* 9 (3) <http://jasss.soc.surrey.ac.uk/9/3/10.html> (Accessed December 5, 2017).

Humphreys, Paul. 2004. *Extending Ourselves: Computational Science, Empiricism, and Scientific Method*. Oxford: Oxford University Press.

Kitcher, Philip. 1990. The Division of Cognitive Labor. *The Journal of Philosophy* 87: 5-22.

Kitcher, Philip. 1993. *The Advancement of Science: Science without Legend, Objectivity without Illusion*. Oxford: Oxford University Press.

Longino, Helen. 1995. Gender, Politics, and the Theoretical Virtues. *Synthese* 104: 383-97.

Martini, Carlo, and Manuela Fernández Pinto. 2017. Modeling the Social Organization of Science: Chasing Complexity through Simulations *European Journal for Philosophy of Science* 7 (2): 221-38.

Miller, John H., and Scott E. Page. 2007. *Complex Adaptive Systems: An Introduction to Computational Models of Social Life.* Princeton Studies in Complexity. Princeton: Princeton University Press.

Muldoon, Ryan, and Michael Weisberg. 2011. Robustness and Idealization in Models of Cognitive Labor. *Synthese* 183 (2): 161-74.

Pöyhönen, Samuli. 2016. Value of Cognitive Diversity in Science. *Synthese*, published online July 1, 1-22.

Sent, Esther-Mirjam. 1997. An Economist's Glance at Goldman's Economics. *Philosophy of Science* 64: 139-48.

Strevens, Michael. 2003. The Role of the Priority Rule in Science. *The Journal of Philosophy* 100: 55-79.

Thoma, Johanna. 2015. The Epistemic Division of Labor Revisited. *Philosophy of Science* 82 (3): 454-72.

Weisberg, Michael. 2013. Modeling Herding Behavior and its Risks. *Journal of Economic Methodology* 20 (1): 6-18.

Weisberg, Michael, and Ryan Muldoon. 2009. Epistemic Landscapes and the Division of Cognitive Labor. *Philosophy of Science* 76 (2): 225-52.

Winsberg, Eric. 2013. Computer Simulation in Science. *Stanford Encyclopedia of Philosophy* <https://plato.stanford.edu/entries/simulations-science/> (Accessed December 5, 2017).

Woodward, Jim. 2006. Some Varieties of Robustness. *Journal of Economic Methodology* 13 (2): 219-40.

Ylikoski, Petri. 1995. The Invisible Hand and Science. *Science Studies* 8: 32-43.

Ylikoski, Petri, and Emrha Aydinonat. 2014. Understanding with Theoretical Models. *Journal of Economic Methodology* 21 (1): 19-36.

Zollman, Kevin. 2010. The Epistemic Benefit of Transient Diversity. *Erkenntnis* 72 (1): 17-35.